

This listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims:**

1. (Currently Amended) A method for monitoring a query during runtime, said query involving a plurality of join operations, the method comprising the steps of:
  - running the query according to a first join order;
  - generating a first portion of a result set for the query while running the query according to the first join order, including adding to the result set a first record that matches the plurality of join operations in the query;
  - concurrent with running the query, collecting performance statistics about each of the join operations;
  - dynamically changing the first join order, during running of the query, to a second join order based on the statistics; and
  - generating a second portion of the result set for the query while running the query according to the second join order, including adding to the result set a second record that matches the plurality of join operations in the query such that the result set includes at least the first and second records, the first and second portions of the result set generated for the same execution of the query.
2. (Canceled)
3. (Previously Presented) The method according to claim 1, further comprising the step of:
  - collecting additional statistics about each of the join operations after the first join order is changed to the second join order.
4. (Original) The method according to claim 3, further comprising the step of:
  - changing the second join order to either the first join order or a third join order based on the additional statistics.

5. (Previously Presented) The method according to claim 1, wherein the plurality of join operations include:

- a first join that includes a first table and a second table;
- a second join that includes the first table and a third table.

6. (Currently Amended) The method according to claim 5, further comprising the steps of:

- determining respective fan-in statistics for the first join and second join; and
- dynamically changing the first join order to ~~a~~ the second join order if the respective fan-in statistics indicate that the second join is more likely to cause fan-in than the first join.

7. (Currently Amended) The method according to claim 5, further comprising the step of:

- determining respective fan-out statistics for the first join and the second join; and
- dynamically changing the first join order to ~~a~~ the second join order if the respective fan-out statistics indicate that the second join is less likely to cause fan-out than the first join.

8. (Currently Amended) The method according to claim 5, further comprising the steps of:

- determining respective fan-in statistics for the first join and second join;
- determining respective fan-out statistics for the first join and the second join; and
- dynamically changing the first join order to ~~a~~ the second join order based on a combination of the respective fan-in and fan-out statistics.

9. (Currently Amended) The method according to claim 1, comprising the steps of:

- identifying a predetermined sample size;

performing the step of collecting statistics for the predetermined sample size;  
evaluating the collected statistics; and  
dynamically changing the first join order to a the second join order based on the collected statistics.

10. (Original) The method according to claim 9, comprising the steps of:  
collecting additional statistics for substantially all of the query;  
comparing the additional statistics with the collected statistics; and  
adjusting the predetermined sample size, for use by a subsequent query,  
according to results of the comparing step.

11. (Previously Presented) The method according to claim 1, further comprising the steps of:  
running another query after the query; and  
selecting an initial join order for the other query based on the collected performance statistics.

12. - 23. (Canceled).